

**CLAIMS**

1           1.       A method for dynamically patching code, comprising the steps of:  
2           intercepting program instructions;  
3           determining if a program instruction is to be replaced; and  
4           dynamically replacing the program instruction with a replacement instruction .

1           2.       The method of claim 1, wherein the step of dynamically replacing the  
2           program instruction comprises fetching a replacement instruction and storing it in a  
3           code cache.

1           3.       The method of claim 2, wherein the step of dynamically replacing the  
2           program instruction further comprises executing the replacement instruction in lieu of  
3           the program instruction each time a function associated with the program instruction  
4           is required.

1           4.       The method of claim 3, wherein the replacement instruction comprises  
2           part of a patch that is made available via an application programming interface.

1           5.       The method of claim 1, further comprising the step of, prior to  
2           determining if a program instruction is to be replaced, determining if the program  
3           instruction has been cached.

6. The method of claim 5, further comprising the step of executing the cached instruction in lieu of the program instruction if an associated instruction has been cached.

7. The method of claim 1, further comprising the step of, prior to intercepting program instructions, gaining control over execution of program instructions by injecting a dynamic execution layer interface into the program.

8. The method of claim 1, further comprising the step of dynamically receiving information about program instructions to be replaced and replacement instructions to replace the program instructions.

9. The method of claim 1, further comprising the step of executing transition code if a program instruction to be replaced is currently running.

10. A system for dynamically patching code, comprising:  
 means for gaining control over execution of a program;  
 means for intercepting program instructions;  
 means for determining if a program instruction is to be replaced with a new instruction; and  
 means for dynamically replacing the program instruction with a replacement instruction .

11. The system of claim 10, wherein the means for dynamically replacing the program instruction comprise means for fetching a replacement instruction and storing it in a code cache.

12. The system of claim 10, further comprising means for determining if a program instruction has been cached.

13. The system of claim 10, further comprising means for dynamically receiving information about program instructions to be replaced and replacement instructions that are configured to replace the program instructions.

14. A dynamic patching program stored on a computer-readable medium, comprising:

- logic configured to gain control over execution of a program;
- logic configured to intercept program instructions;
- logic configured to determine if a program instruction is to be replaced; and
- logic configured to dynamically replace the program instruction with a replacement instruction .

15. The program of claim 14, wherein the logic configured to dynamically replace the program instruction comprises logic configured to fetch a replacement instruction and store it in a code cache.

16. The program of claim 14, further comprising logic configured to determine if a program instruction has been cached.

1           17.     The program of claim 14, further comprising logic configured to  
2     dynamically receive information about program instructions to be replaced and  
3     replacement instructions that are configured to replace the program instructions.

1           18.     A method for dynamically patching code, comprising the steps of:  
2             gaining control over the execution of a program;  
3             intercepting program instructions;  
4             determining whether the program instructions have been cached and, if so,  
5     executing the cached instructions;  
6             if the program instructions have not been cached, determining if the program  
7     instructions are to be replaced; and  
8             dynamically replacing the program instructions with replacement instructions  
9     if it is determined that the program instructions are to be replaced.

1           19.     The method of claim 18, wherein the step of dynamically replacing the  
2     program instructions comprises fetching replacement instructions and storing them in  
3     a code cache.

1           20.     The method of claim 19, wherein the step of dynamically replacing the  
2     program instructions further comprises executing the replacement instructions in lieu  
3     of the program instructions each time a functionality associated with the program  
4     instructions is required.

1           21.    The method of claim 20, wherein the replacement instructions  
2   comprise part of a patch that is made available via an application programming  
3   interface.

1           22.    The method of claim 18, further comprising the step of executing  
2   transition code if a program instruction to be replaced is currently running.